



Analizando excepciones

Transcripción

[00:00] Bienvenidos nuevamente. Entonces y continuando donde nos quedamos, hemos visto que un error puede causar que el programa automáticamente se cierre. Esa es la primera cosa que que nos estamos dando cuenta. Él no puede terminar sus instrucciones debido a que esta excepción salió.

[00:20] Segunda cosa que tenemos que ver es que dependiendo del error, Java tiene un nombre para cada uno de ellos. En el caso que tuvimos aquí fue `ArithmeticException`, que ya nos da una buena idea de qué se trata. Y aquí un pequeño mensaje, `slash by zero`, que quiere decir división entre 0.

[00:44] Entonces esto es una de las partes buenas de Java, porque otros lenguajes, por ejemplo para especificarte los errores que ocurren, usan lo que es un diccionario de errores. Él simplemente hace un crash, él no termina ejecución y te dice: "error código tal", entonces tú tienes que ir a internet o la documentación del lenguaje que estás usando y buscar por el código de error y recién ahí darte una idea pues de qué está pasando.

[01:10] En el caso de Java, la excepción, ya tiene esa información de qué es lo que pasó. Hay tipos de excepciones, hay tipos de errores. Entonces justo que estamos tocando este tema de los tipos de errores, vamos a entrar a nuestra `claseCuenta`, por ejemplo. Y en `claseCuenta` tenemos nuestro método `saca`.

[01:34] Y aquí en `saca`, pues nuestra razón para retornar verdadero o falso es si es que tienes saldo o no. Pero en el mundo real hay muchas más razones por

las que tú no podrías sacar dinero. Por ejemplo, tu cuenta puede estar desactivada, puede ser un indicio de fraude, no tienes saldo, hay varios motivos por los cuales tú no podrías sacar dinero del banco.

[02:02] ¿Y yo como clasificó cada error para esos motivos? Por ejemplo, ¿yo cómo sé que si es que no tenía saldo te retornó falso? ¿Pero yo como sé que ese error que hubiera lanzado aquí ese error fue porque no tenía saldo? ¿De qué manera yo clasifico eso? Las excepciones también son objetos. Es primera cosa que tenemos que entender en este punto.

[02:28] Una excepción como tal también es un objeto que tiene su nombre y tiene su mensaje. Entonces, eso es lo que nos va a ayudar a entender y a crear hasta nuestras propias excepciones, de acuerdo a los errores que nosotros queremos controlar también. El objetivo básico de la excepción es explicar el error que ha ocurrido.

[04:54] Es dar una explicación más humana sobre qué error exactamente ha ocurrido. Volvemos aquí a nuestro flujo. Aquí vemos pues que, en efecto él imprimió una vez la primera línea, llegó aquí, imprimió 1, y después Arithmetic Exception. Yo podría lanzar aquí otra excepción X, y el daría otro tipo de excepción.

[03:19] ¿Cómo hago entonces, para trabajar con estos distintos tipos de errores que yo ya les he mostrado? En el siguiente video vamos a descubrir cómo.